TPC Simulation Update

Carlos Perez (SBU)

TPC framework

This all is done after Geant4 simulation: will not affect the incoming sim production

Simulation: Hits to **data**

Current Alan's approach
- Diffusion
- Distortions

Diffusion was patched by me two months ago to solve issue of diff-constants & Bfield

New approach (under dev, not yet merged)
- Diffusion
- Distortions
- Modularization
- GEMResponse
- PadResponse
- TimeResponse

Reconstruction: **data** to clusters

Current Clusterizer (patched by me two months ago)
- peak finder
- peak fitter
- Estimation of errors and size of cluster (**solely from data distributions**)

New additions (under dev, not yet merged)
- Charge charing compensation
- PulseTime tailing
- PulseTime disentangle

Distribution of (clusterX - hitX)

Current issue reported by Haiwang, Sourav and Tony is in width of distribution for arc difference reported by evaluator.

To reproduce the problem I turn off SCdistortions and had a look at the distributions for single pions in [-0.5, 0.5] eta with 2GeV momentum. In fact I saw 80um width in these gaussian looking distributions.

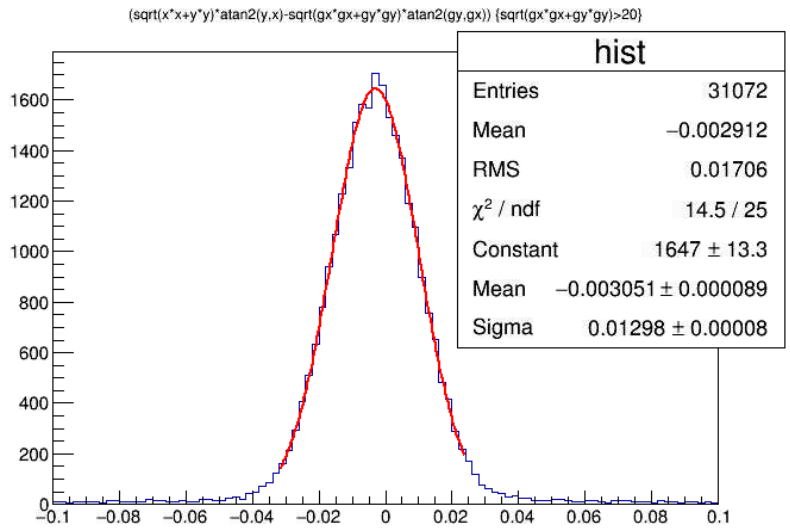
I check at the g4eval directory. Cluster-hit matching is done based on nearest peak matching, not centre of gravity: width is not taken into account when generating these plots. Since truth-PR and truth-fitter uses the evaluator, it is easier to do a peak smearing in the sim-2-data section.

I added and tested the following code:

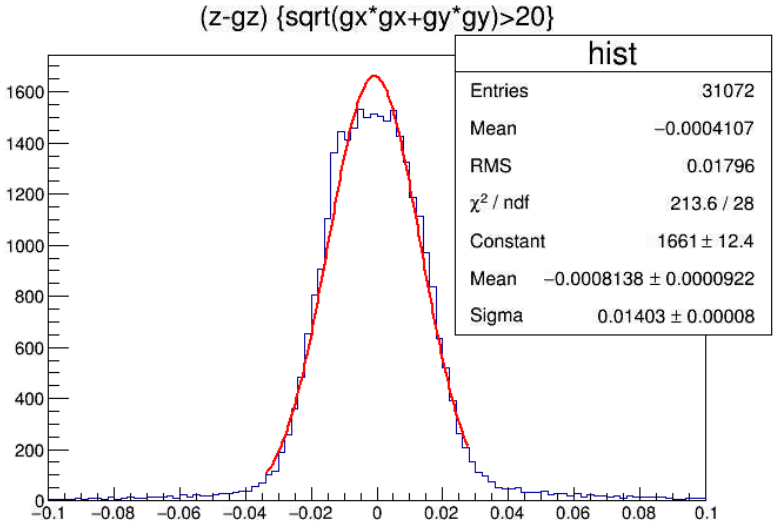
```
@@ -343,6 +344,26 @@ int PHG4CylinderCellTPCReco::process_event(PHCompositeNode *topNode)
        double sigmaL = 0.010; //100um
         double cloud_sig_rp = sqrt( fDiffusionT*fDiffusionT*(fHalfLength - TMath::Abs(hi
         double cloud_sig_zz = sqrt( fDiffusionL*fDiffusionL*(fHalfLength - TMath::Abs(hi
+
+       //===============
+       // adding a random displacement to effectively deal with cellularisation
+       float fFractRPsm = 0.09;
+       float fFractZZsm = 0.06;
+       phi += fFractRPsm*rand.Gaus(0,cloud_sig_rp)/r;
+       if(phi>+TMath::Pi()) phi -= TMath::TwoPi();
+       if(phi<-TMath::Pi()) phi += TMath::TwoPi();
+       z += fFractZZsm*rand.Gaus(0,cloud_sig_zz);
+       // moving center
+       phibin = geo->get_phibin( phi );
+       zbin = geo->get_zbin( z );
+       // bin protection
+       if(phibin < 0 || phibin >= nphibins){continue;}
+       if(zbin < 0 || zbin >= nzbins){continue;}
+       // bincenter correction
+       phidisp = phi - geo->get_phicenter(phibin);
+       zdisp = z - geo->get_zcenter(zbin);
+       //=====
+
        int n_rp = int(3*cloud_sig_rp/(r*phistepsize)+1);
         int n_zz = int(3*cloud_sig_zz/zstepsize+1);
```

TPC sim+clus Results

400 pions

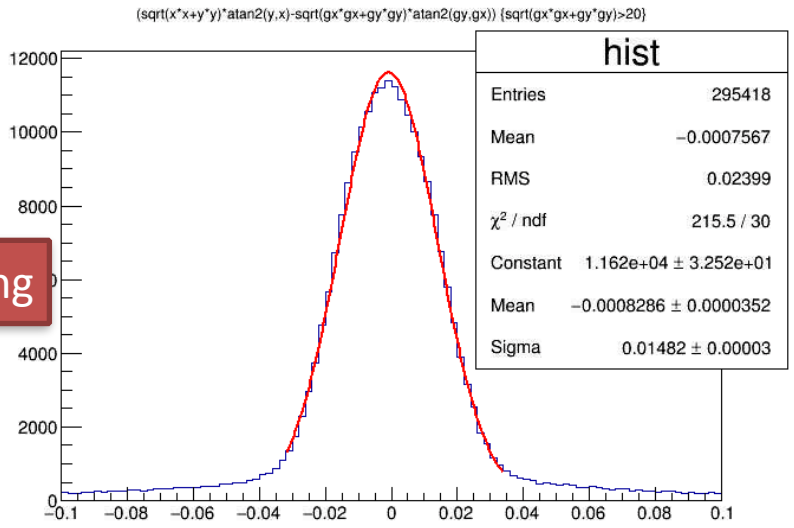


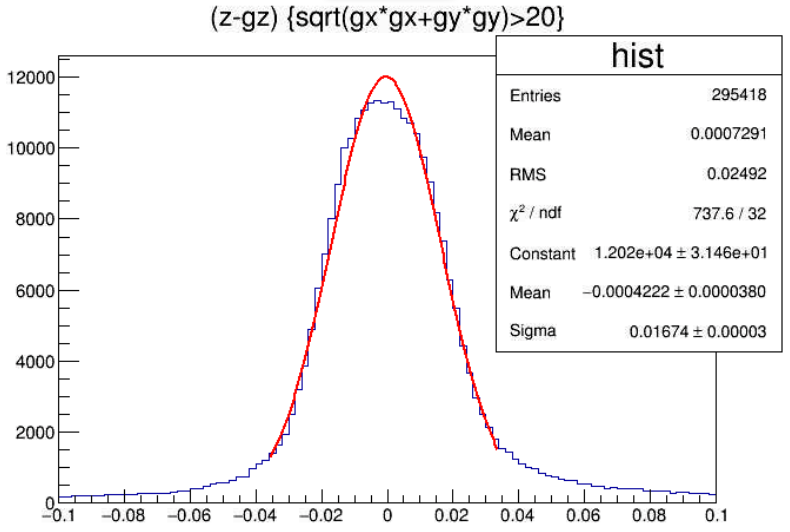
rphi - grphi



z - gz

Central Hijing





patch

- If you agree, I can add setters and getters to use these optional fine tuners (or turned them off) at macro level.
- Keep in mind, that this will effectively will not change the error estimation at reconstruction, since that depends solely on the data distribution around the maxima (not biased by MC match)

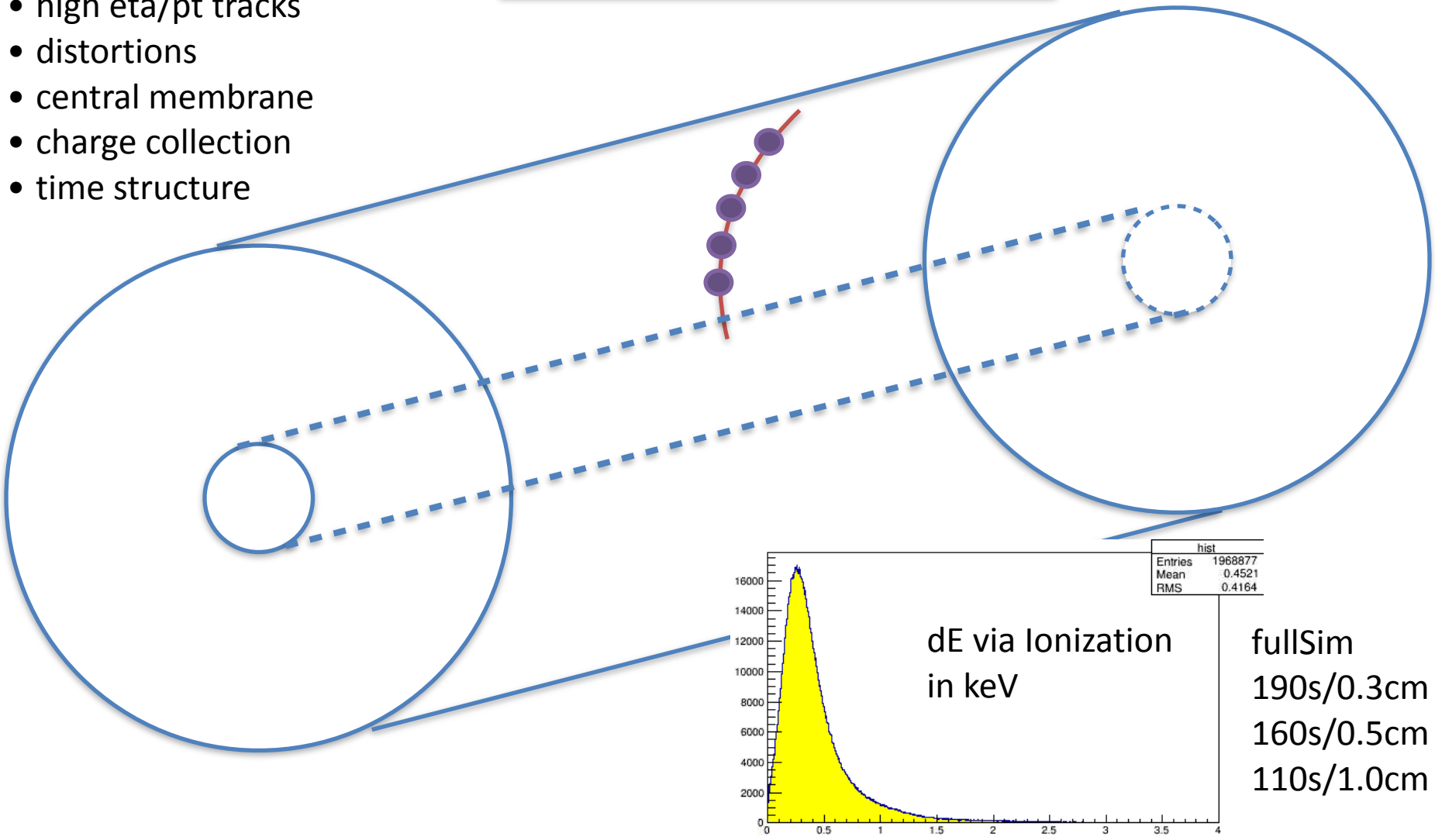
- As mentioned several times, the current approach is using an effective approach to the response of the TPC to particles crossing it. The new approach (under development) will include all these features from first principles so it will need no hacks (at least I hope so =)

BACKUP

One big active volume

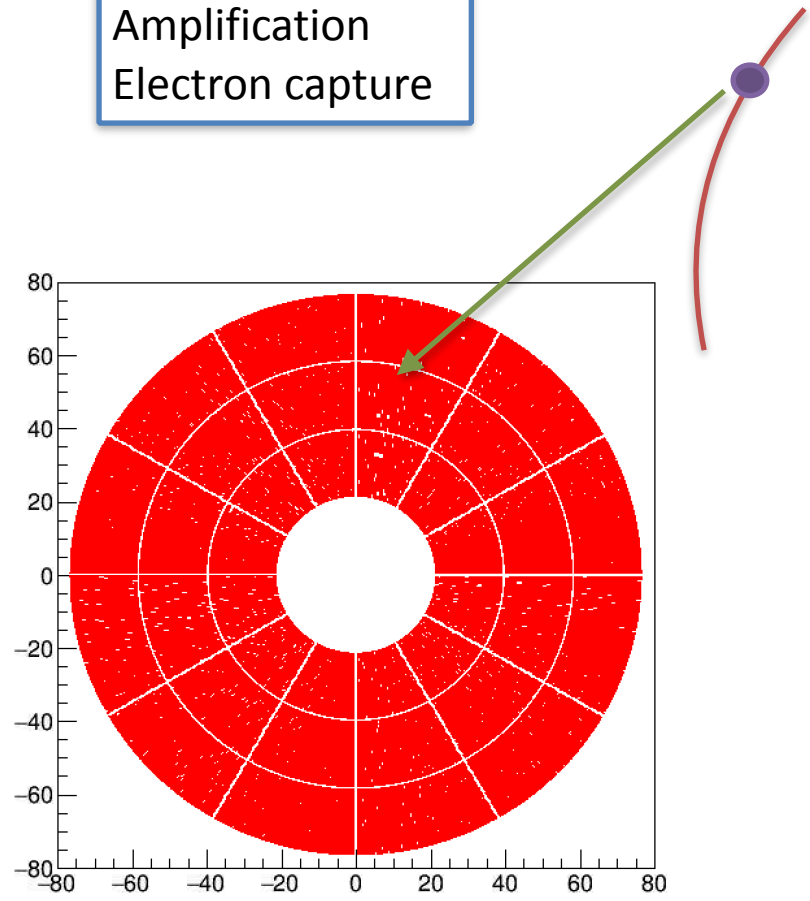
Better description of
- high eta/pt tracks
- distortions
- central membrane
- charge collection
- time structure

StepLength is controlled via Geant4



TPC Simulation starts after G4 hit

Transport
Distortions
Amplification
Electron capture



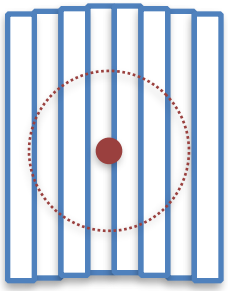

```
List of Nodes in Fun4AllServer:
Node Tree under TopNode TOP
TOP (PHCompositeNode)/
   DST (PHCompositeNode)/
      PHG4INEVENT (PHDataNode)
      PHHepMCGenEvent (IO,PHHepMCGenEvent)
      PIPE (PHCompositeNode)/
         G4HIT_PIPE (IO,PHG4HitContainer)
      SVTX (PHCompositeNode)/
         G4HIT_SVTX (IO,PHG4HitContainer)
         SvtxHitMap (IO,SvtxHitMap_v1)
         SvtxClusterMap (IO,SvtxClusterMap_v1)
         SvtxTrackMap (IO,SvtxTrackMap_v1)
         SvtxVertexMap (IO,SvtxVertexMap_v1)
      TPCHits (IO,PHObject)
      MAGNET (PHCompositeNode)/
         G4HIT_MAGNET (IO,PHG4HitContainer)
      G4HIT_BH_1 (IO,PHG4HitContainer)
      BH_FORWARD_PLUS (PHCompositeNode)/
         G4HIT_BH_FORWARD_PLUS (IO,PHG4HitContainer)
      BH_FORWARD_NEG (PHCompositeNode)/
         G4HIT_BH_FORWARD_NEG (IO,PHG4HitContainer)
      G4TruthInfo (IO,PHG4TruthInfoContainer)
      BBC (PHCompositeNode)/
         BbcVertexMap (IO,BbcVertexMap_v1)
      G4CELL_SVTX (IO,PHG4CellContainer)
      TPCDigits (IO,PHObject)
      GLOBAL (PHCompositeNode)/
         GlobalVertexMap (IO,GlobalVertexMap_v1)
```

still used to keep framework intact

new internal consumable

new internal consumable

Pad Matching



sketch of pads in traverse plane and cloud from microsimulation

Cloud is projected into RO geometry. Algorithm computes range of pads compatible with cloud centroid and spread and return range of pairs (PAD;QUOTA)

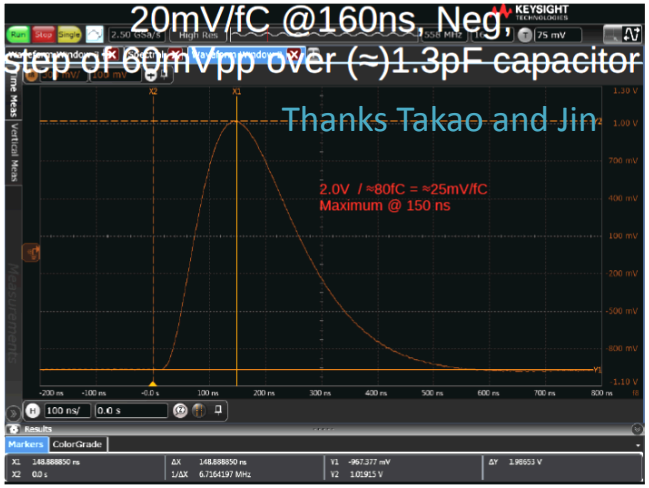
Quotas are computed using PDF in cylindrical coords. CDF is obtained by integrating in radius to cover layer and simplifying based on small angle approximation.

Example of two hits left by track from Central Hijing ev.

```
PushCloud2Module 56
CLOUD WEIGHT 30000 to be stored in 8 pads.
  PAD 1129(8) weight 92.076 to be stored in 4 time bins.
  PAD 1130(8) weight 1332.66 to be stored in 5 time bins.
  PAD 1131(8) weight 1084.78 to be stored in 5 time bins.
  PAD 1132(8) weight 48.7581 to be stored in 4 time bins.
  PAD 1257(9) weight 932.654 to be stored in 5 time bins.
  PAD 1258(9) weight 12120.9 to be stored in 7 time bins.
  PAD 1259(9) weight 8855.97 to be stored in 6 time bins.
  PAD 1260(9) weight 353.867 to be stored in 5 time bins.
  ==> SUM PAD WIGHTS 24821
PushCloud2Module 56
CLOUD WEIGHT 16000 to be stored in 6 pads.
  PAD 1130(8) weight 47.9372 to be stored in 3 time bins.
  PAD 1131(8) weight 20.6066 to be stored in 3 time bins.
  PAD 1257(9) weight 1238.98 to be stored in 6 time bins.
  PAD 1258(9) weight 8547.83 to be stored in 6 time bins.
  PAD 1259(9) weight 3292.65 to be stored in 6 time bins.
  PAD 1260(9) weight 65.2591 to be stored in 5 time bins.
  ==> SUM PAD WIGHTS 13213
PushCloud2Module 56
```

Pulse shape

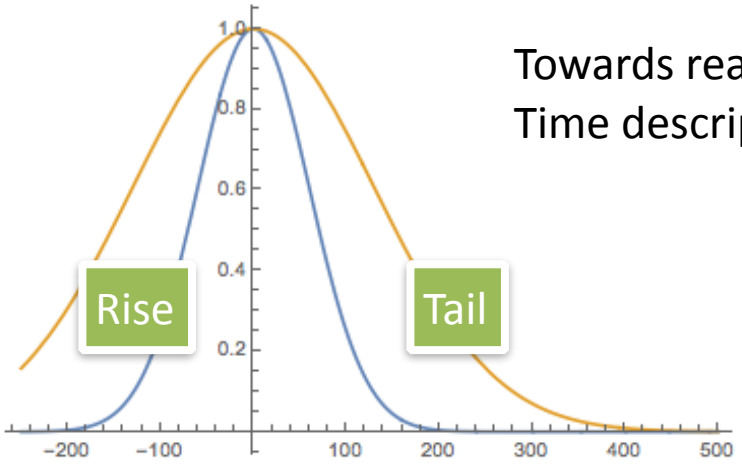
TPC Shaping time measurement



Example of sampling at 10 MSPS

```
CLOUD WEIGHT 8000 to be stored in 11 pads.
  PAD 458(3) weight 164.358 to be stored in 8 time bins.
    TIME 288 weight 3
    TIME 289 weight 32
    TIME 290 weight 58
    TIME 291 weight 29
    TIME 292 weight 21
    TIME 293 weight 11
    TIME 294 weight 5
    TIME 295 weight 1
    ==> SUM TIME WIGHTS 160
  PAD 459(3) weight 840.758 to be stored in 9 time bins.
    TIME 288 weight 17
    TIME 289 weight 164
    TIME 290 weight 300
    TIME 291 weight 151
    TIME 292 weight 107
    TIME 293 weight 60
    TIME 294 weight 26
    TIME 295 weight 9
    TIME 296 weight 2
    ==> SUM TIME WIGHTS 836
```

```
Plot[{Exp[-0.5 (x)^2/60^2], Exp[-0.5 (x)^2/130^2]}, {x, -250, +500}
 PlotRange -> All]
```



Towards realistic
Time description

Possible use time shape to deconvolute occupancy?

Status of this development

- New simulation incorporates crucial aspects of the TPC not addressed yet: dEdX, GEM amplification and Pulse Shape

- Incorporated into Fun4All framework: subsystems and input/output
- From G4Hits to TPCDigits all done: In-gas transport, GEM, Digitizer, rectangular RO pads matching
- Clusterer not yet ready. Needs fine-tune to account for pulse shape
- All contained in parallel library (does not interfere with current Svtx libs)
- 20 new files ready: TPCConstants TPCDataTypes TPCDigit TPCPadMapTPCDigitsContainer TPCCloud TPCDetector TPCDetectorSubsystem TPCEventAction TPCSimulation TPCSimulationSubsystem TPCSteppingAction